

# TER : Bras Robot pour jeu de plateau

Berriri Mehdi, Bounab Fayçal, Djema Sofiane, Doglio Arthur

June 2019

# Sommaire

|      |                                         |    |
|------|-----------------------------------------|----|
| I/   | Introduction . . . . .                  | 2  |
| 1/   | Présentation du groupe . . . . .        | 2  |
| 2/   | Présentation du sujet . . . . .         | 2  |
| II/  | État de l'art . . . . .                 | 3  |
| 1/   | Contexte . . . . .                      | 3  |
| 2/   | Technologies utilisées . . . . .        | 4  |
| III/ | Travail effectué . . . . .              | 6  |
| 1/   | Études . . . . .                        | 6  |
| 2/   | Hardware . . . . .                      | 12 |
| 3/   | Software . . . . .                      | 14 |
| 4/   | Client et Serveur . . . . .             | 21 |
| IV/  | Gestion de projet . . . . .             | 22 |
| 1/   | Organisation . . . . .                  | 22 |
| 2/   | Tâche . . . . .                         | 23 |
| 3/   | Problèmes rencontrés . . . . .          | 24 |
| V/   | Conclusion . . . . .                    | 24 |
| VI/  | Perspectives et réflexions personnelles | 25 |
| 1/   | Perspectives . . . . .                  | 25 |
| 2/   | Réflexions personnelles . . . . .       | 26 |
| VII/ | Annexes . . . . .                       | 27 |
| 1/   | Journal de bord . . . . .               | 27 |

# **I/ Introduction**

Dans le cadre de notre de notre Master 1 Informatique nous avons souhaité prendre comme travail d'étude et de recherche le sujet qui porte le nom "Bras Robot pour jeu de plateau". Nous avons choisi ce sujet parce que nous pensons que les robots sont un excellent moyen pour expérimenter l'apprentissage par les essais et les erreurs. Nous avons choisis le jeu d'échec comme jeu de plateau pour le projet.

## **1/ Présentation du groupe**

Notre groupe est composé de : Berriri Mehdi, Bounab Fayçal, Djema Sofiane, Doglio Arthur et nos intervenants sont : M Renevier Philippe et Mme Pelleau Marie.

## **2/ Présentation du sujet**

Le but de ce TER est de développer une librairie de commande du bras robot Joy-It ROBOT02, ou d'améliorer une librairie existante.

Cela afin de mettre en oeuvre quelques mouvements dans un jeu réel, en fixant les emplacements ou en utilisant une librairie de reconnaissance pour les éléments d'un jeu.

## II/ État de l'art

### 1/ Contexte

Notre bras est composé de six Servo-moteurs, deux d'entre eux font des rotations (le Servo 0 qui sert de plate-forme et le Servo 4 qui fait tourner la pince).

Le numéro 5 permet l'ouverture et la fermeture de la pince, et les Servos 1, 2 et 3 permettent l'articulation du bras pour la prise ou la pose d'une pièce d'échec.

Ce bras évolue dans un repère 3D (x, y, z).

Nous avons commencé par le calibrage.

Nous avons démonté chacun des Servos du robot afin de fixer ses positions initiales et son orientation.

Pour cela nous avons calibré chacune des parties du bras c'est-à-dire que nous avons fait en sorte que les différents moteurs aient la plus large amplitude de mouvement pour prendre ou poser des objets au sol.

Nous avons également pris de multiples mesures pour trouver par exemple les différentes distances entre les moteurs, les limites d'angles de chaque moteur ainsi que les équivalences entre les angles réels et ceux de la librairie (disponible dans le document "PriseDeMesures" sur github).

Nous sommes ensuite passé à la partie "mathématique" du TER c'est-à-dire calculer les différents angles de chacune des articulations afin d'atteindre une coordonnée donnée.

Nous avons implémenté plusieurs méthodes. Une permettant le calcul des angles de chaque cerveau en prenant en compte des coordonnées (x, y, z), une faisant la conversion d'un nombre de degrés en radians, une méthode de déplacement des articulations ainsi qu'une méthode gérant l'ordre de mouvement des Servos.

## 2/ Technologies utilisées

L'environnement de développement d'Arduino :



Nous utilisons l'environnement de développement d'Arduino. Le langage de programmation utilisé est le C++, compilé avec avr-g++ 9 et lié à la bibliothèque de développement Arduino, permettant l'utilisation de la carte et de ses entrées/sorties. La mise en place de ce langage standard rend aisé le développement de programmes sur les plates-formes Arduino à toute personne maîtrisant le C ou le C++.

### Arduino Uno Wi-fi REV2 :



Nous utilisons une carte Arduino qui est la base de la programmation avec microcontrôleur (un « mini-ordinateur » qui se présente sous la forme d'un circuit intégré).

Il existe de nombreuses versions de cette carte. Nous avons commencé sur une carte Arduino Leonardo, puis nous avons changé pour une carte Arduino Uno wifi rev2 qui nous a permis d'intégrer un serveur.

### Bras Joy-it ROBOT02 :



Nos encadrants nous ont imposé le robot JOY-IT ROBOT02, ce bras robuste livré avec servomoteurs, pince et visserie de montage.

Le bras est équipé de 6 puissants servomoteurs de déplacement, un à la base du bras pour l'orientation, 3 pour l'articulation (épaule, coude et poignet), un pour l'orientation de la pince et un pour la pince elle-même.

La valeur de ce bras est de 176,40 euros

### III/ Travail effectué

#### 1/ Études

##### Arduino et Raspberry

Nous avons fait une comparaison entre nos arduino (Leonardo et Uno Wifi REV2) et le raspberry Pi.

Voici un descriptif des cartes électroniques sur lesquelles nous avons travaillé.

Arduino Leonardo :



La Leonardo est sortie après la carte Arduino Uno, et a été dotée d'un nouveau type de processeur, et d'une ergonomie revue. Elle a le même nombre de ports que la carte Arduino Uno (12 analogiques et 20 numériques dont 7 pwm). Elle a l'avantage de posséder un port USB, ce qui permet par exemple de brancher un clavier ou une souris.

Cette carte est très bien pour débiter un projet de domotique, comme la gestion de l'éclairage de sa maison ou la fermeture/ouverture des volets en fonction de la météo extérieure.

Nous pouvons l'utiliser bien entendu sur des projets de niveau intermédiaire.

### Arduino Uno Wifi REV2 :



Cette Arduino possède (14 broches d'entrée/sortie numériques (dont 6 peuvent être utilisées comme sorties PWM), 6 entrées analogiques, un résonateur céramique de 16 MHz, une connexion USB, une prise d'alimentation, un en-tête ICSP et un bouton de réinitialisation.

Elle comporte tout ce qui est nécessaire à la prise en charge du microcontrôleur.



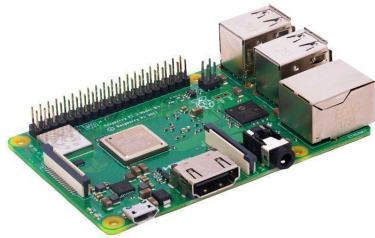
Il suffit de la connecter à un ordinateur à l'aide d'un cordon USB ou de l'alimenter avec un adaptateur c.a./c.c. ou une batterie. Le module Wi-Fi est un SoC autonome avec pile de protocole TCP/IP qui peut donner accès au réseau Wi-Fi (Ou l'appareil peut agir en tant que point d'accès).

Une caractéristique utile de Uno Wi-Fi est la prise en charge de la programmation OTA (over-the-air), que ce soit pour le transfert d'esquisses Arduino ou de firmware Wi-Fi.

Il comprend un nouveau microprocesseur 8 bits à partir de Microchip.

Il est doté d'un IMU (unité de mesure d'inertie). Nous pouvons nous connecter au réseau Wi-Fi avec le module wifi intégré. La connexion wifi est sécurisée par un accélérateur de puce cryptée.

#### Raspberry Pi 3 Model B+ :



Le raspberry possède un processeur quad-core 64 bits de 1,4 GHz, LAN sans fil à double bande, Bluetooth 4.2/BLE, Ethernet plus rapide et prise en charge de l'alimentation sur Ethernet.

Pour conclure, étant donné que sur raspberry, la programmation se fait avec le langage Python et que nous avons aussi besoin d'un écran ou bien de lancer une machine virtuel pour avoir accès à l'interface de celle-ci, nous avons préféré rester avec une carte arduino.

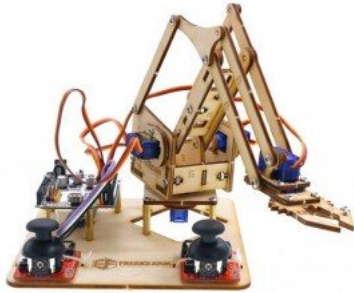
De plus, notre choix a été conforté par le fait que nous avons déjà beaucoup travaillé sur le langage d'Arduino qui est le c++.

## **Bras Robotique**

Nous avons fait une comparaison entre notre bras (Joy-it ROBOT02) et d'autres bras robotiques potentiels.

Voici un descriptif de plusieurs bras robotiques avec des gammes de différents niveaux.

Bras robotisé en kit FreaksArm EF08070 :



C'est un bras motorisé didactique commandé via une carte Freaduino compatible Arduino. Il est à assembler soi-même et est composé d'un châssis en bois, de servomoteurs permettant des mouvements sur 4 axes et de 2 joysticks.

La valeur de ce bras est de 75,90 euros.

### Bras Dobot Basic :



Le bras robotisé et modulaire Dobot Basic, destiné au milieu éducatif, permet de s'initier à la robotique, à la programmation et à la conception 3D.

Ce bras accepte de nombreuses applications comme la réalisation de dessins, l'impression 3D et le déplacement d'objets.

Les éléments permettant d'obtenir toutes ces fonctionnalités sont inclus dans le kit et sont facilement intégrables au bras.

Le châssis est en aluminium anodisé et en plastique ABS assurant une bonne robustesse.

Le conception de ce bras permet d'obtenir des opérations avec une précision de 0,2 mm grâce à ses 4 axes et ses différents moteurs.

Il peut être contrôlé de plusieurs façons via un PC par cordon USB, par bluetooth (smartphone et tablette, application Android et iOS disponible) et peut être programmé en C++, C, Python et Java.

La valeur de ce bras est de 1215 euros.

### Bras robotique Braccio T050000 :



Le bras motorisé et modulaire Braccio d'Arduino est livré en kit à assembler soi-même et se contrôle via une carte Arduino ou compatible non incluse.

Ce bras peut effectuer différents mouvements grâce à un ensemble de servomoteurs et dispose d'une pince pouvant saisir des objets d'un poids allant jusqu'à 150 g.

Le bras Braccio est polyvalent et sa modularité lui permet d'obtenir différentes configurations.

La valeur de ce bras est de 238 euros.

Pour conclure, nous avons fait une étude sur les différents bras robotiques avec lesquels nous aurions pu travailler.

Le FreakArm est trop simple pour réaliser des mouvements dans notre projet.

Le bras Dobot est très bien pour pousser notre projet au maximum de ces capacités mais le prix est 10 fois plus cher que celui de notre bras actuel.

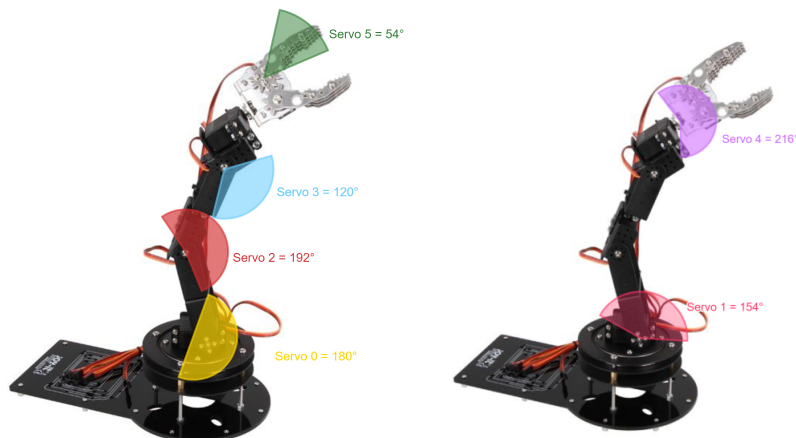
Pour finir il y a le bras Braccio qui est deux fois plus cher que notre bras robot et qui serait également idéal pour ce projet. Le prix est donc l'élément le plus important que nous avons pris en compte pour choisir notre modèle de bras robot.

## 2/ Hardware

### Assemblage et calibrage

Nous avons eu deux opérations importantes à faire sur le bras, son assemblage ainsi que son calibrage. Celui-ci nous a été fourni assemblé mais nous l'avons démonté et remonté entièrement afin de calibrer les 6 Servos. La phase de calibrage nous a donc permis de définir tous les mouvements possibles pour chaque articulation du bras.

Les amplitudes des Servos :



## Prise de mesures

Suite à cela nous avons pris différentes mesures : les distances entre chaque Servos à l'aide d'une règle et ainsi que leurs angles de rotation à l'aide d'un rapporteur. Nous avons ensuite confirmé nos mesures grâce à des photos et un rapporteur virtuel.

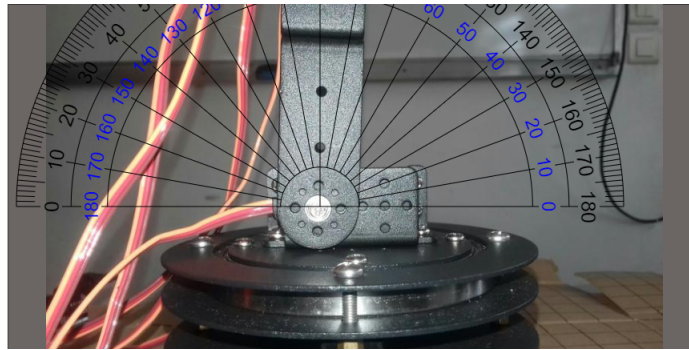


FIGURE 1 – Prise de mesure

Les mesures des angles et distances :

| Moteur | Valeur Min | Valeur Max | Valeur par degré | Mesure valeur/degré | Ouverture Pince (cm) |
|--------|------------|------------|------------------|---------------------|----------------------|
| Servo0 | 90         | 580        | 2.45             | 580 --> 200°        |                      |
|        |            |            |                  | 90 --> 0°           |                      |
|        |            |            |                  | 335 --> 100°        |                      |
| Servo1 | 90         | 475        | 2.5              | 475 --> 0°          |                      |
|        |            |            |                  | 90 --> 154°         |                      |
|        |            |            |                  | 225 --> 100°        |                      |
|        |            |            |                  | 250 --> 90°         |                      |
| Servo2 | 100        | 580        | 2.5              | 580 --> 197°        |                      |
|        |            |            |                  | 100 --> 5°          |                      |
|        |            |            |                  | 312 --> 90°         |                      |
|        |            |            |                  | 337 --> 100°        |                      |
| Servo3 | 90         | 390        | 2.5              | 390 --> 130°        |                      |
|        |            |            |                  | 90 --> 10°          |                      |
|        |            |            |                  | 315 --> 100°        |                      |
| Servo4 | 90         | 570        | 100/45           | 570 --> 216°        |                      |
|        |            |            |                  | 90 --> 0°           |                      |
|        |            |            |                  | 290 --> 90°         |                      |
| Servo5 | 190        | 310        | 100/45           |                     | 190 --> fermé        |
|        |            |            |                  |                     | 200 --> 1.4cm        |
|        |            |            |                  |                     | 250 --> 5.2cm        |
|        |            |            |                  |                     | 300 --> 3.5cm        |
|        |            |            |                  |                     | 310 --> 9.4cm        |

| Distances(cm)   |      |
|-----------------|------|
| Servo 1 à 2     | 10,4 |
| Servo 2 à 3     | 9,8  |
| Servo 3 à pince | 14   |

Nous avons également mesuré les dimensions du jeu d'échecs utilisé.

Les mesures du jeu d'échec :

| (en cm)           | Position du plateau |           |         |
|-------------------|---------------------|-----------|---------|
| Centre Case A8    | { 19,1 ; 10,5 }     |           |         |
| Centre Case H8    | { -20,8 ; 10,5 }    |           |         |
| Taille d'une case | 5,7                 |           |         |
| Pince (en cm)     | Ouverture           | Fermeture | Hauteur |
| fou               | 4,5                 | 1,5       | 3,5     |
| dame              | 4,5                 | 1,8       | 5       |
| tour              | 4,5                 | 1,8       | 4       |
| plon              | 3,5                 | 1,5       | 3       |
| cavalier          | 4,5                 | 1,8       | 2       |

### 3/ Software

#### Classes

Nous avons deux classes.

Une pour le plateau de jeu ayant en attribut une représentation de l'échiquier, le nombre de pièces adverses prises ainsi qu'une méthode resetPlateau().

L'autre classe Servo possède les attributs suivant :

```
int numServo; // port de la carte Arduino sur lequel est branché le Servo
int mini; // valeur minimum du Servo
int maxi; // valeur maximum du Servo
float coeffDir; // coefficient directeur de la fonction calculMoveDegre
int valeurPos; // valeur en position initiale
int initAngle; // angle minimum du Servo
int vitesse; // vitesse de rotation du Servo
```

et trois méthodes :

- `int calculMoveDegre(int angle)` : calcul la valeur à laquelle le Servo doit se positionner pour former l'angle donné en paramètre.
- `void moveValeur(int valeur)` : déplace le Servo à la valeur donnée en paramètre.
- `void setVitesse(int v)` : permet de modifier la vitesse de rotation du Servo.

## Méthodes de calcul

Voici les schémas du bras sur lesquelles nous nous sommes basé pour calculer les emplacement de chaque servos :

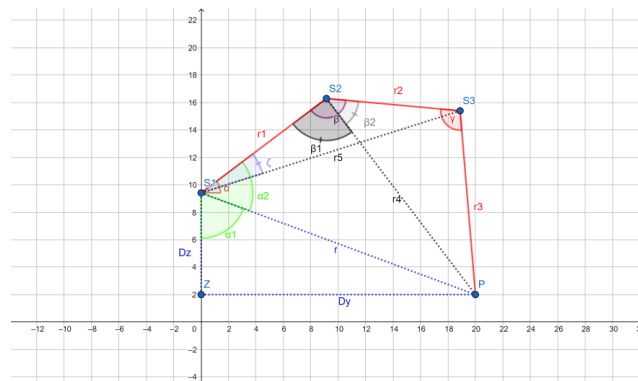


FIGURE 2 – Vue de profil du bras





Cette methode fonctionne de la maniere suivante :

Tout d'abord, il faut orienter le bras vers sa destination. Pour cela, nous calculons la valeur de l'angle du Servo 0.

```
r6 = sqrt(x * x + y * y);
r7 = sqrt(r6 * r6 - r0 * r0);
omega1 = acos(x / r6);
omega2 = (PI / 2) - acos(r0 / r6);
omega = omega1 - (omega2 / 2);
```

Il faut maintenant déterminer les valeurs des angles des Servos formant l'articulation du bras. Pour cela nous utilisons la Formule d'al-Kashi (ou loi des cosinus) à plusieurs reprises.

Pour cela, nous fixons l'angle du Servo 1 à 130 degrés (commencer au delà est inutile) puis nous calculons les angles que doivent prendre les Servos 2 et 3 pour arriver à destination.

Nous répétons cette opération pour un angle du Servo 1 décroissant de 5 en 5 jusqu'à 0.

```
Dy = abs(y - S1y);
Dz = abs(z - S1z);
r = sqrt(Dy * Dy + Dz * Dz);

if(z <= S1z){ alpha1 = asin(Dy / r);}
if(z > S1z){ alpha1 = (PI / 2) + asin(Dz / r);}
alpha = degreToRadian(Dalpha);
alpha2 = alpha + (PI / 2) - alpha1;

r4 = sqrt(r1 * r1 + r * r - 2 * r * r1 * cos(alpha2));
beta1 = acos((r1 * r1 + r4 * r4 - r * r) / (2 * r1 * r4));
beta2 = acos((r2 * r2 + r4 * r4 - r3 * r3) / (2 * r2 * r4));
beta = beta1 + beta2;
gamma = acos((r2 * r2 + r3 * r3 - r4 * r4) / (2 * r2 * r3));
```

Enfin, la combinaison d'angles que nous conservons est celle où le Servo 3 est le plus perpendiculaire à la coordonnées de destination.

```

r5 = sqrt(r1*r1 + r2*r2 - 2 * r1 * r2 * cos(beta));
zeta = acos(((r1*r1+r5*r5 - r2*r2)/(2*r1*r5)));
S3y = cos(alpha - zeta) * r5;
if(abs(S3y - y) < abs(distance - y) && angleS2 > 10 && angleS2 < 200 && angleS3 > 10
    distance = S3y;
    AS0 = angleS0;
    AS1 = angleS1;
    AS2 = angleS2;
    AS3 = angleS3;
}

```

Dans le fichier main, nous avons le serveur et deux méthodes

- mouvementBras(float x, float y, float z, float ouverturePince, float fermeturePince, int typeMouvement) : Si typeMouvement est égal à 0 cette méthode permet de prendre une piece. Dans le cas contraire elle posera une piece.
- jouer(String instruction) : La méthode jouer permet d'effectuer un coup d'échec passé en paramètre

Nous avons essayé de déplacer plusieurs Servos en même temps, mais suite à différentes tentatives, nous nous sommes rendu compte que cela était impossible à cause d'un temps de calcul nécessaire trop important.

Nous activons donc les Servos les uns après les autres.

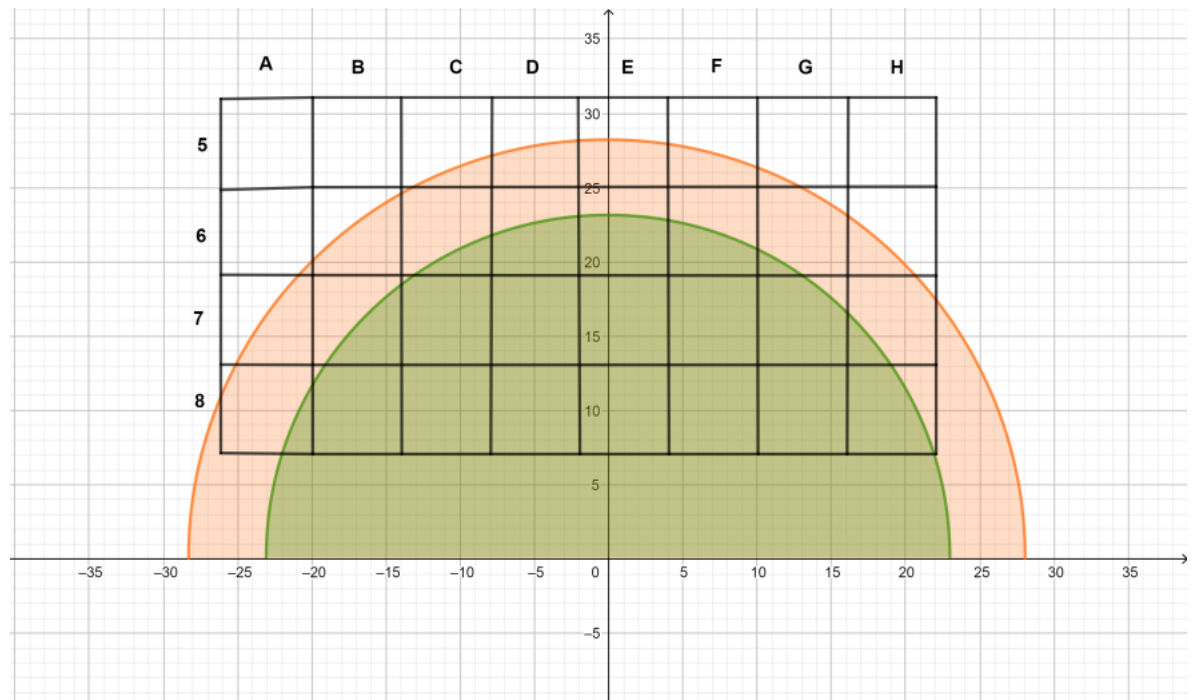
Voici la manière dont s'effectue le mouvement du bras :  
 Tout d'abord, nous ouvrons la pince si c'est une prise de pièce.  
 Ensuite on active le Servo 0 pour orienter le bras. Puis on active les Servos 3 et 2 pour préparer la descente vers la destination.  
 Enfin c'est le Servo 1 qui s'active pour baisser le bras.  
 Une fois arrivé à destination, il faut ouvrir ou fermer la pince.  
 Enfin, on active le Servo 1 pour remonter le bras afin de le placer au dessus du plateau.

## Tests

Nous avons testé la portée du bras en déplaçant les différentes pièces d'une position proche vers une position de plus en plus éloignée.

Nous avons donc mesuré une portée maximale de 28 centimètres. Cependant, au delà de 23 centimètres, la dernière équerre du bras n'est plus perpendiculaire au plateau ce qui fait que les pièces ne se posent pas correctement.

Schéma de la portée du bras :



Pour la précision, l'objectif était que les pièces soient posées sans qu'elles ne dépassent des cases de notre plateau.

Nous avons donc déplacé chaque pièce 10 fois entre 2 cases du plateau se trouvant dans la partie verte du schéma précédent. A chaque déplacement les pièces étaient correctement posées dans leurs cases.

En faisant ce même test avec une des deux cases se trouvant dans la partie orange, la pièce n'était pas posé correctement et tombait avant la fin des dix déplacements.

Nous avons donc une précision de plus ou moins 1,5 centimètre. Ce résultat s'explique par le fait que lorsque nous donnons un angle à un Servo, il a une précision de plus ou moins deux degrés.

Nous avons ensuite mesuré la vitesse de rotation des Servos grâce à la fonction `millis()` d'arduino.

Nous avons calculé le temps avant et après le déplacement, puis nous avons mesurer la différence des deux.

Ces temps sont utilisés pour la valeur de l'attribut vitesse des Servos qui est comprise entre 0 et 12.

Les résultats obtenus sont les suivant :

| vitesse | Servos 0,1,2,3 (degrés/s) | Servo 4,5(degrés/s) |
|---------|---------------------------|---------------------|
| 0       | 400                       | 500                 |
| 1       | 200                       | 250                 |
| 2       | 133                       | 167                 |
| 3       | 100                       | 125                 |
| 4       | 80                        | 100                 |
| 5       | 67                        | 83                  |
| 6       | 57                        | 71                  |
| 7       | 50                        | 62                  |
| 8       | 44                        | 56                  |
| 9       | 40                        | 50                  |
| 10      | 36                        | 45                  |
| 11      | 33                        | 42                  |
| 12      | 31                        | 38                  |

Nous avons choisi la valeur 10 comme vitesse pour nos Servos, car elle permet au bras de ne pas basculer à cause de sa propre vitesse.

#### 4/ Client et Serveur

Le bras robot est contrôlé grâce à un client web. Au démarrage de l'Arduino celle-ci se connecte à un réseau et acquiert une IP. Lorsqu'un navigateur web se connecte à l'adresse de l'IP la carte Arduino lui renvoie un code HTML.

Avec le client, on peut sélectionner une suite de coups et l'envoyer à l'Arduino. Lorsque le coup est envoyé le bras robot effectue sa tâche, pendant ce laps de temps le client s'actualise. La page finie de s'actualiser lorsque le coup est joué. A la fin de l'actualisation, le plateau s'affiche à nouveau sur le client.

L'interface client :



L'interface est constituée des cases du plateau sur lesquelles il faut cliquer. Les cases sélectionnés sont affichées dans le zone de texte à droite.

Une fois que deux cases sont choisies, il faut appuyer sur le bouton "Envoyer" pour jouer un coup. Si on se trompe, on peut appuyer sur le bouton "Effacer" et sélectionner à nouveau le coup que l'on veut effectuer.

## IV/ Gestion de projet

### 1/ Organisation

Pour l'organisation nous nous voyions de façon hebdomadaire le jeudi et vendredi aux campus des lucioles afin de travailler ensemble et discuter de la progression du Ter.

Si notre objectif hebdomadaire n'était pas atteint, nous continuions chez nous et nous communiquions via l'application Discord. Nous organisions aussi des rendez-vous avec nos encadrants toutes les trois semaines dans le but de montrer nos résultats et demander conseil pour les semaines à venir. De plus nous tenions un journal de bord chaque semaine ,disponible sur notre git, afin que nos encadrants soient informés du travail que nous faisons chaque semaine.

Dans le but de mieux s'organiser et d'avoir une trace écrite des difficultés rencontrées nous avons mis en place un dépôt sur Git Hub.

Notre dépôt est composé d'un dossier "src" dans lequel se trouve le dossier "pronation" contenant le code permettant d'utiliser le bras, un dossier "Documents" contenant le journal de bord, le fichier sur les études et un fichier excel avec des tableaux sur nos tests.

Voici le lien du Github :

<https://github.com/faycalbounab/TER-Bras-Robot>

## 2/ Tâche

Voici la répartition du travail pour chaque membre du groupe (Arthur (A) , Fayçal (F) , Mehdi (M) , Sofiane (S)) :

- Calibrage du bras (**A,M,S**)
- Prise des mesures (**M,S,F**)
- Implementation d'une interface client (**A,F**)
- Implementation d'un serveur (**A,S**)
- Découpage du code (classes et main) (**M,F**)



### 3/ Problèmes rencontrés

- L'impossibilité de lancer un code dû à des problèmes de localisation du fichier source par rapport au dossier des librairies.
- Le bras effectuait des mouvements brusques et incontrôlables à chaque rotation de Servo. Nous avons réglé ce problème en maîtrisant la vitesse de ces rotations.
- La pince qui se desserre à chaque prise. Pour régler cela nous avons rajouté deux vis au niveau des deux crochets de la pince.
- Nous avions un décalage lors de la prise des pièces qui était causé par le fait que l'orientation du bras était calculé par rapport à l'arrête droite du bras.
- La carte arduino doit être branchée à un ordinateur pour pouvoir l'alimenter suffisamment pour qu'une connexion avec un serveur-client ait lieu. Nous pensons que ce problème peut être réglé grâce à une alimentation adéquat.

### V/ Conclusion

Dans le cadre de nos dernières années de licence en informatique nous avons eu l'occasion de faire plusieurs projets de programmation, tous différents les uns des autres.

Pour notre première année de Master nous avons choisi de prendre un sujet qui combine et touche des domaines différents (Mathématique et Robotique).

C'est l'aspect de découverte qui nous attirait dans ce projet. Nous voulions des nouveaux outils qui pourraient enrichir notre expérience dans le domaine de l'informatique.

Lorsque nous avons débuté ce projet, nous avons des difficultés à anticiper la charge de travail mais cela nous a aidés à mieux nous organiser et travailler de façon plus efficace

Grâce aux notions que nous avons acquises ces dernières années, nous avons trouvé que la partie software était abordable. Notre bonne cohésion d'équipe nous a aidé à effectué le côté un peu complexe de l'interaction hardware-software (bien calibrer le bras, faire des mouvements précis au bras...).

Pour conclure, le sujet que nous avons choisi nous a vraiment permis de mettre en application nos connaissances en informatique tout en obtenant de nouvelles notions.

Dans le futur, il sera plus facile pour nous de refaire des projets impliquant un lien hardware-software.

## VI/ Perspectives et réflexions personnelles

### 1/ Perspectives

Pour aller plus loin dans le projet, nous pourrions améliorer la portée du bras en ajoutant des équerres plus grandes (même si nous pensons qu'il pourrait y avoir des problèmes de répartition du poids).

Il serait également possible d'ajouter d'autres fonctionnalités utiles à un jeu de plateau comme un lancer de dé.

Nous également pu faire jouer le bras de façon autonome en ajoutant un ou plusieurs capteurs transmettant l'état du plateau au bras (et pourquoi pas implementer une IA pour le robot).

## 2/ Réflexions personnelles

- **Berriri Mehdi** : J'ai trouvé le projet passionnant autant du point de vue des découvertes de la robotique que de la programmation Arduino. J'ai aimé développer en Arduino et découvrir cet outil. Ce projet m'a permis d'appliquer mes connaissances en mathématiques et informatique à un projet qui me tient à coeur.
- **Bounab Fayçal** : Ce Ter m'a personnellement énormément apporté, notamment grâce au travail que j'ai effectué sur la partie hardware et software. Au début ce n'était pas facile mais à force de redoubler d'efforts je ressors plus expérimenté et plus aptes à faire des projet sur des longues durées.
- **Djema Sofiane** : J'ai trouvé ce projet très passionnant dans le sens où il m'a fait découvrir une facette de l'informatique qui ne m'intéressait pas à première vue et je parle bien sûre de la robotique. Le fait d'avoir un résultat visuel de ce qu'on a codé est une expérience assez satisfaisante et si je devais retravailler sur un projet de ce type, ce serait un grand plaisir de le faire.
- **Doglio Arthur** : Je suis du même avis que mes collègues en ce qui concerne la découverte de la robotique. Ce projet était très intéressant par son aspect multidisciplinaire. Avec ce dernier, j'ai pu revoir et réutilisé mes connaissances en géométrie. J'ai aussi appris beaucoup de choses qui m'ont permis de mieux comprendre les interactions hardware-software.

## VII/ Annexes

### 1/ Journal de bord

**Semaine du 04/03/2019** Nous avons récupéré le bras robotique auprès de nos encadrants, nous allons commencer par calibrer le bras avant de réfléchir aux jeux que nous allons étudier. Nous pensons déjà travailler sur le jeu de société Takenoko et le jeu d'échecs. Dans ce TER nous ferons une étude sur les différents bras robotiques que nous aurions pu avoir et voir aussi la différence entre un arduino et une raspberry.

**Semaine du 11/03/2019** Cette semaine nous avons démonté et remonté tous les moteurs du bras robotique afin de les calibrer. Après avoir fini l'assemblage nous avons remarqué que ses mouvements étaient trop brusques de ce fait nous avons diminué la vitesse de ses mouvements. Pour la semaine prochaine on essayera de lui faire prendre un objet quelconque pour avoir une vision plus précise des possibilités offertes par le robot.

#### **Semaine du 19/03/2019**

Cette semaine nous avons réussi à faire attraper et reposer une pièce du jeu d'échecs sans la faire retomber. Pour la semaine prochaine on essaiera de faire des calculs sur les angles pour faire en sorte que les différentes parties (surtout les parties gérées par le moteur 2,3, 4) du robot ne se gênent pas lors du mouvement.

**Semaine du 25/03/2019** Cette semaine nous avons préparé un quadrillage pour positionner en  $(x, y)$  le bras dans l'espace. Puis, nous avons commencé à calculer les angles de déplacement des moteurs et leur position initiale dans l'espace. Pour la semaine prochaine nous continuerons sur les calculs afin de les implémenter.

**Réunion nos encadrants** Nous avons fait une démonstration du bras robotique, il a ainsi fait une prise d'une pièce d'échec. Ensuite nous avons vu avec Monsieur Rennevier s'il y a la possibilité d'intégrer le Wi-Fi à notre carte arduino. Nous allons faire une recherche sur la carte Wi-Fi à commander et sur les différences entre les raspberrys et notre carte arduino.

**Semaine du 01/04/2019** Cette semaine nous avons fini les calculs des angles(voir fichier Angles), nous avons commencé un système de contrainte(formules) pour avoir le champ d'action du bras robotique. Pour le découpage du code, nous avons fait une classe servo pour avoir des objets qui correspondent à chacun des Servos.

**Semaine du 08/04/2019** Cette semaine nous avons fait un bilan avec Monsieur Malapert, nous avons testé des formules qui permettent de calculer les angles afin d'effectuer un mouvement avec le bras. Nous intégrerons ces formules au code par la suite. Nous allons aussi faire une étude en parallèle sur la possibilité de passer sur Raspberry.

**Semaine du 15/04/2019** Cette semaine nous avons mis en ligne les calculs via un fichier HTML, qui renvoie vers une page où l'on peut préciser les coordonnées dans l'espace(x, y). Il y a aussi en ligne le début de l'étude des bras robotiques. Nous avons recalibré le Servo 4. Enfin nous avons intégré une fonction de calcul des angles de rotation des 4 premiers Servos.

**Semaine du 22/04/2019** Cette semaine nous avons fait deux fonctions qui renvoient les angles de rotation des 4 premiers Servos sous forme d'un tableau. Nous lisons les coordonnées x, y et z en entrée standard que nous utilisons pour effectuer un mouvement.

**Semaine du 29/04/2019** Cette semaine nous avons implémenté la prise et la pose d'une pièce (on rentre des positions pour la position de la pronation, le bras se déplace vers la position indiquée, il prend la pièce et se met en position initiale puis attend les nouvelles coordonnées qui indiquent la position où il doit déposer la pièce). Une diminution des mouvements incontrôlables lors du 1er mouvement. Et un moyen de contrôler la vitesse des Servos.

**Semaine du 06/05/2019** Cette semaine nous avons implémenté deux approches permettant de calculer les mouvements que doivent effectuer les Servos du bras pour se déplacer vers une coordonnée (x, y, z).

**Réunion avec nos encadrants** Nous avons présenté les mouvements de prise et pose d'un fou, cependant il y avait encore des imprecisions causées par une méthode de calcul des déplacements qui n'était pas aboutie. Nous avons discuté d'une possible augmentation de la portée du bras grâce à de plus grandes équerres. Nous avons aussi dit que nous avons l'intention de travailler sur le contrôle à distance du bras grâce à une raspberry et l'arduino que Mr Renevier nous a déposé. Mme Pelleau nous a aussi laissé un module Wi-Fi pour arduino.

**Semaine du 13/05/2019** Nous avons recalibré les Servos 2 et 3. Nous commençons le travail sur l'arduino Wi-Fi et nous avons essayé de faire en sorte de bouger plusieurs Servos en même temps mais cela n'est pas possible car le temps de calcul nécessaire est trop important.

**Réunion avec nos encadrants** Nous avons présenté les mouvements de prise et pose d'un fou. Nous avons parlé du jeu d'échecs, nous allons faire jouer le bras sur un plateau de taille 5x5. Nous essayons d'activer plusieurs Servos en même temps et faire fonctionner le Wi-Fi. Mr Rennevier nous a conseillé de tester les limites du bras ( précision et portée), quelles informations intégrées dans le rapport (calibrage du bras) et comment réaliser une vidéo pour la présentation (durée de la vidéo et présenter une partie en peu de coups).

**Semaine du 20/05/2019** Cette semaine, nous avons repris des mesures plus précises des rotations des Servos (en prenant des photos et avec un rapporteur virtuel). Nous avons testé l'arduino Wi-Fi qui fait fonctionner le bras correctement et nous avons commencé à faire un client pour celle-ci. Nous ferons une pause la semaine prochaine.

**Semaine du 03/06/2019** Cette semaine, nous avons commencé le rapport et nous avons commis une première version du rapport, il n'est pas complet il manque encore des informations que nous aurons d'ici à la semaine du 10/06. Pour le moment c'est surtout pour avoir une idée très large de comment il sera. Nous avons aussi fini les calculs de notre bras pour la précision de celui-ci grâce à une méthode que nous avons faite et nous sommes toujours entrain de bosser en parallèle le wi-fi.

**Réunion avec nos encadrants** Nous avons présenté notre client et notre serveur qui est pour le moment opérationnel il nous reste à l'intégrer à notre code afin de tester, nous avons présenté une première version du rapport et pour finir nous avons montré l'amélioration des mouvements du bras.

**Semaine du 10/06/2019** Cette semaine nous avons finalisé le projet, nous nous sommes occupés des derniers détails à régler, faire la vidéo de démonstration, finaliser le rapport et mettre en place les diapositives.